AIAA Scitech 2021 Forum, 11–15 and 19–21 January 2021, Online Event
Paper number AIAA 2021-0175
https://doi.org/10.2514/6.2021-0175

# Development of UCAV Fleet Autonomy by Reinforcement Learning in a Wargame Simulation Environment

Burak Yuksek*
*DeltaV Space Technologies Inc., Istanbul, 34906, Turkey*

Mustafa Umut Demirezen†
*Roketsan Missiles Inc., Ankara, 06780, Turkey*

Gokhan Inalhan‡
*Cranfield University, United Kingdom, MK43 0AL*

**In this study, we develop a machine learning based fleet autonomy for Unmanned Combat Aerial Vehicles (UCAVs) utilizing a synthetic simulation-based wargame environment. Aircraft survivability is modeled as Markov processes. Mission success metrics are developed to introduce collision avoidance and survival probability of the fleet. Flight path planning is performed utilizing the proximal policy optimization (PPO) based reinforcement learning method to obtain attack patterns with a multi-objective mission success criteria corresponding to the mission success metrics. Performance of the proposed system is evaluated by utilizing the Monte Carlo analysis in which a wider initial position interval is used when compared to the defined interval in the training phase. This provides a preliminary insight about the generalization ability of the RL agent.**

## I. Introduction

In aerial attack and defense scenarios, combat aircrafts fly through the man-made hostile environment to perform surveillance, reconnaissance, and neutralization of ground and air targets. The hostile environment may include several defense units such as antiaircraft artilleries (AAA), surface-to-air missiles (SAM) and detection and tracking radar systems. All of these units are selected and placed according to strategical importance and geographical specifications of the defended area and characteristics of the possible threats that may attempt to attack high-value targets. Generally, a seamless air defense system is generated by using these separate ground units to increase kill probability of the threats. From the attacker aircraft point of view, the main aim is to destruct the strategical targets such as enemy armoury, command base and other assets. In this mission, the aircraft will probably be detected and tracked by enemy radars and engage with ground-to-air defense systems. Beside of the mission accomplishment, it is also a critical factor to come back to the air base safely, i.e. unharmed, through the hostile environment to provide sustainability of the mission. For this reason, flight path planning step has a crucial importance to generate a feasible and safe flying route which increases the mission success in the threat area.

Combat survivability of the aircraft is defined as its ability to avoid or withstand a hostile environment [1]. It aims to evaluate the survival probability of the aircraft and provides quantitative measure about the path safety. The survivability of the UCAVs directly depends on susceptibility and vulnerability level which are functions of ability of the aircraft to avoid incoming threats and withstanding a hostile environment, respectively. The susceptibility characteristics is related with intensity of the enemy air defense systems, aircraft basic design (i.e. low radar cross-section, smokeless engine, etc.), survivability equipment (ordnance, onboard electronic attack equipment, etc.) and aircraft tactics. The vulnerability characteristics are influenced by enemy's warhead type, aircraft basic design (i.e. fuel tank position, etc.) and survivability equipments (i.e. armour of the fuel tanks, adaptive flight control systems to compensate any control degradation) that reduces the impact of the warhead when aircraft is hit. In a man-made hostile environment, there may be radar systems, SAM and AAA weapons and no-fly zones in which it is forbidden to fly. An example warfare map is illustrated in Figure (1). Here, the aim of the aircraft is to infiltrate into the target area (command center), destroy it by using a guided munition and fly back to the air base.

*R&D Engineer, byuksek@deltav.com.tr, AIAA Professional Member.
†Head of Artificial Intelligence and Technology Management Unit, umut.demirezen@roketsan.com.tr.
‡BAE Systems Chair, Professor of Autonomous Systems and Artificial Intelligence, inalhan@cranfield.ac.uk, AIAA Associate Fellow.

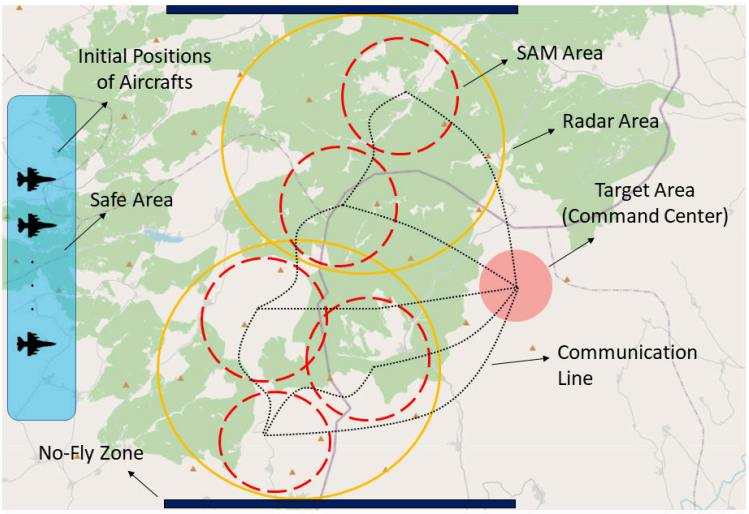


**Fig. 1 General view of an example warfare.**

Several methods and models are developed to analyse the survivability of the combat missions. In [2], a survivability model is developed for aircraft in 5-states Markov model structure, i.e. undetected, detected, tracked, engaged and hit. Probabilities of each states are calculated during the mission which provide the information about the survivability of the aircraft. This model also captures the behavior of the enemy defense system elements which can communicate with each other. In [3], two air mission routes are compared from an air combat survival perspective by utilizing this 5-state Markov model. In [4], 5-state survivability model is enhanced by introducing uncertainty regarding the locations of enemy radar and weapon systems. Uncertainty effects on the survivability prediction is evaluated by Monte Carlo simulations and mean and standard deviation are calculated for the expected survivability level. In [5], an analytic model is developed for engagement level aircraft survivability based on the stochastic duel theory. This model represents the encounter process which includes target detection, acquisition, firing and reloading steps. Beside of providing the survival probability of the aircraft, the survivability contour maps can be obtained with the mean time of detection, single shot kill probability and other parameters by using the proposed model. In [6], 6-state Markov model is developed to evaluate the torpedo effectiveness in pursuit mission of a diesel submarine. Defender's countermeasure and maneuvering capabilities are taken into consideration for both sides. According to the results of this study, it is observed that two pairs of jammer with two decoys is a proper combination to saturate a torpedo attack against the submarine which can provide information to decision makers about the engagement behaviors. These studies focused on survivability analysis of the aircrafts while not taking into account the survival probability data to solve the flight path planning problem.

Beside of the route evaluation, survivability models are also used in the flight path planning process in which it is desired to complete the combat mission while maximizing the survival probability. In [7], survivability of an unmanned vehicle is defined by using damage and hit probability in the hostile environment. Cost function is developed and used in extended $A^*$ algorithm to find the optimal path. The performance of the proposed system is evaluated in the simulation environment which includes one agent. In [8], mission path planning is performed by using survivability model and reinforcement learning method for an aircraft which flies in a man-made hostile environment. The survivability model consists of 3-state radar and 2-state weapon Markov models. The weapon model has multiple-shot capability which effects the kill probability of the aircraft. Several cost functions are defined to calculate the total cost which is used as reward in the training phase. Then, optimal and safe route for the aircraft is generated by using reinforcement learning method called Deep Q-Learning. In these studies, one-to-many combat scenarios are studied in which one attacker tries to survive in a hostile environment with more than one defender enemy assets. They did not expand the flight path planning application for UCAV fleet cooperative attack scenarios to saturate the defense units.

In this study, we proposed to develop an RL agent as a centralized mission planner to find the collision-free and safe flight route. A 5-state survivability model is used which includes search, detection, tracking, engagement and hit states, to evaluate the kill probability of the combat aircraft [2]. Transition matrices are defined separately for three different areas; i.e. outside of the radar area, inside of the radar area and inside of the weapon area. According to the mission

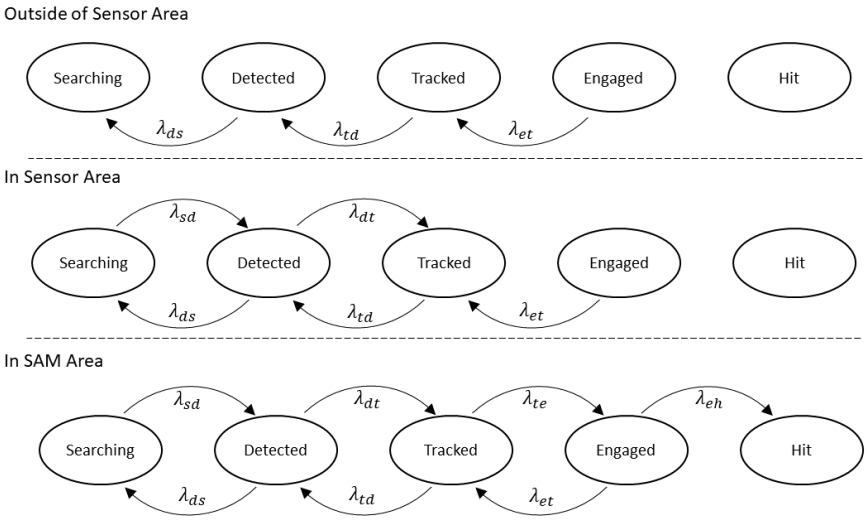


**Fig. 2 5-State survivability model.**

scenario, the attacker aircraft fleet should infiltrate into the target area and destroy them. Several mission success metrics are defined for survival probability of the aircraft fleet to obtain a quantitative data about the operation effectiveness. Path planning is performed by using reinforcement learning (RL) method in which the RL agent has actor-critic structure that generates acceleration and velocity command signals for the UCAVs. Observation vector of the RL agent consists of; a) states of the UCAVs b) relative distance and angle data between the UCAVs and enemy assets such as radars and weapon systems, c) survivability states of the UCAVs. Reward function is developed to evaluate the system performance by considering the survival probability and collision-avoidance ability of the UCAV fleet. In the training phase of the RL agent, Proximal Policy Optimization (PPO) algorithm is utilized. Success metrics are developed and effectiveness of the proposed coordinated flight path planning method is evaluated by using 2000-runs Monte Carlo simulation.

Remaining of this manuscript is organized as follows; in Section II, point mass model of the aircraft and its survivability model is defined. Several preliminary simulations are performed and their results are explained. In Section III, RL-based path planning process is described, RL agent structure is given and observation and reward functions are presented. In Section IV, single-run simulation and 2000-runs Monte Carlo simulation results are evaluated. In Section V, concluding remarks are given and future works are introduced.

## II. Mathematical Models

Performance of the RL agent is directly related with the fidelity level of the mathematical model. Realistic, high-fidelity aircraft and environment model provide accurate observation data for the training phase. If the mathematical model of the environment has adequate level of fidelity, one can expect similar performance with the simulations when the algorithm is implemented in real life application. In this study, we used a simple point-mass mathematical model for ease of implementation of the proposed algorithm under certain assumptions; a) low-level attitude controllers are active and the closed-loop system can be modeled as first-order transfer function, b) velocity and altitude controllers are active and the closed-loop system can be modeled as a first-order transfer function c) disturbance effects can be rejected effectively by the control systems. Time constants of the first-order transfer functions are selected by considering the physical limits of the aerial vehicle to obtain a realistic model. By using these assumptions, point-mass mathematical model of the UCAVs is given in the following subsection.

### A. Point Mass Model of the Aircraft

Point mass models give insight about aircraft kinematics and they are quite suitable for guidance and path planning applications. Mathematical definition of the point mass model is given in Equation (1).

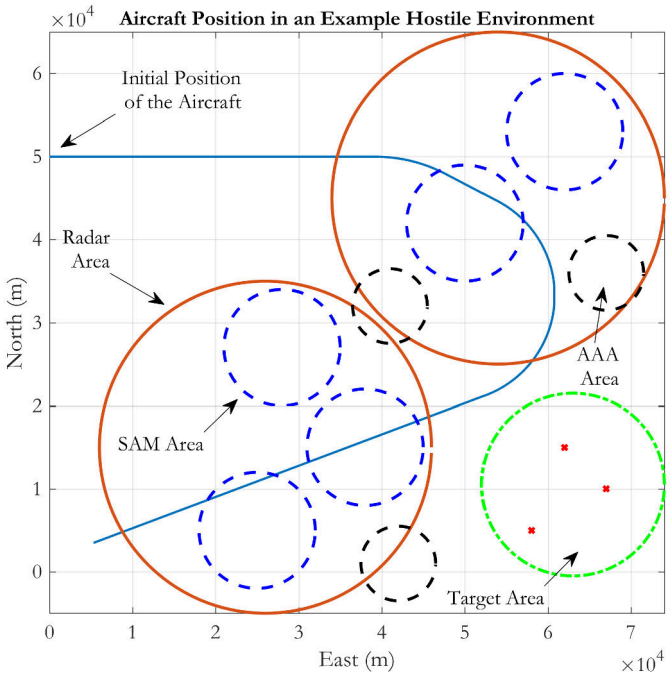


**Fig. 3 Air-to-ground mission example in a hostile environment.**

$$
\begin{aligned}
\dot{p}_n(t) &= V_a(t)\,\cos\psi(t)\,\cos\gamma(t) \\
\dot{p}_e(t) &= V_a(t)\,\sin\psi(t)\,\cos\gamma(t) \\
\dot{h}(t) &= V_a(t)\,\sin\gamma(t) \\
\dot{\gamma}(t) &= b_\gamma\left(\gamma_c(t) - \gamma(t)\right) \\
\dot{V}_a(t) &= b_V\left(V_{a_c}(t) - V_a(t)\right) \\
\dot{n}_{lat}(t) &= b_{n_{lat}}\left(n_{lat_c}(t) - n_{lat}(t)\right)
\end{aligned}
\tag{1}
$$

where $p_n, p_e$ and $h$ are north, east position and altitude. $\gamma, V_a$ and $n_{lat}$ are flight path angle, airspeed, and lateral acceleration. $b_\gamma, b_V$ and $b_{n_{lat}}$ are time constants for closed-loop flight path angle, airspeed and lateral acceleration control systems. Subscript '$c$' defines the commanded signal for the related controlled output.

### B. Survivability Model

Survivability defines the survival probability of the aircraft in the man-made hostile environment. It is modeled by using a continuous-time Markov process, which is a continuous-time, discrete-value stochastic-process for an infinitesimal step size of $\Delta$ [4],

$$
\begin{aligned}
P[X(t+\Delta) = j \mid X(t) = i] &= \lambda_{i,j}(t)\,\Delta \\
P[X(t+\Delta) = i \mid X(t) = i] &= 1 - \textstyle\sum_{j\neq i}^{\infty} \lambda_{i,j}(t)\,\Delta = 1 - \nu_i(t)\Delta
\end{aligned}
\tag{2}
$$

where $\lambda_{i,j}(t)$ is transition intensity and $\nu_i(t)$ is departure rate. The transition intensities are used to define transition rate matrix $\Lambda$ as given in Equation (3).

$$
\Lambda_{ij}(t) = \begin{cases} \lambda_{ij}, & \text{if } i \neq j \\ -\nu_i, & \text{if } i = j \end{cases}
\tag{3}
$$

The state probabilities are described as differential equations as given in Equation (4).

$$
\dot{\mathbf{p}}_s(t) = \Lambda^T(t)\,\mathbf{p}_s(t)
\tag{4}
$$

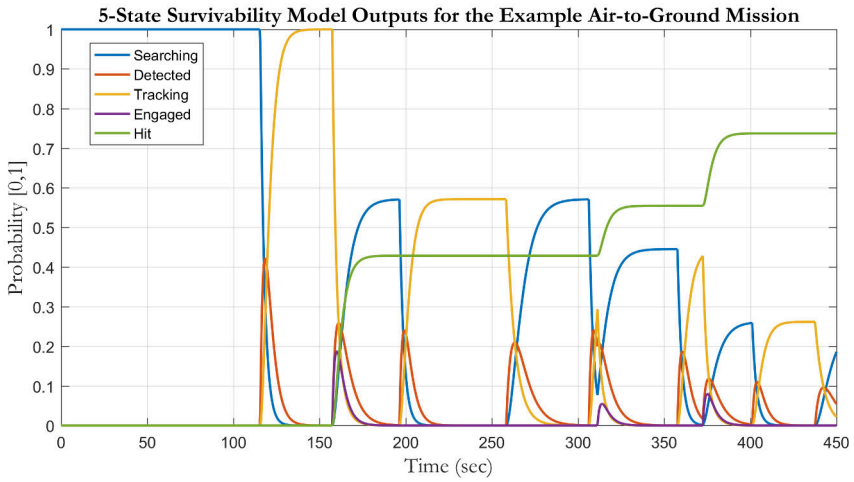


**Fig. 4 Probabilities for the example mission.**

where $\mathbf{p}_s(t)$ describes the state probability vector.

In this study, 5-state survivability model is used which includes un-detected (searching), detected, tracking, engaged and hit states as given in Figure (2) [2]. Transition matrices, which includes transition rates between the states, are related with the position of the aircraft and specific matrices. They are defined for *outside of the radar area, inside of the radar area* and *inside of the weapon area* as ($\Lambda_{outside}, \Lambda_{radar}$ and $\Lambda_{weapon}$), respectively. In this 5-State Markov process, it is also possible to integrate decoy effects which decrease the susceptability and increase the survivability of the target. In an encounter scenario, if the aircraft uses the decoy such as chaff, flare and electronic countermeasures, the transition rate $\lambda_{eh}$ decreases and survival probability of the aircraft increases. Radar cross section (RCS) area of the aircraft determines stealth ability of the platform which is directly related with the transition rates of $\lambda_{sd}$ and $\lambda_{ds}$. These effects can be modeled in the simulation environment for several countermeasures and aircrafts with different stealth characteristics.

To demonstrate the survivabilitiy analysis of a UCAV, an example flight path is illustrated in Figure (3). In this example, the UCAV passes through the radar and weapon areas as a part of a mission solution. For this mission, the time histories of the survivability model states (i.e. searching, detected, tracking, engaged, hit) are given in Figure (4). Here, the UCAV passes through 2 radar areas and 3 SAM areas. As the UCAV passes through these enemy areas, tracking and hit probabilities increase. These time histories provide the survivability level of the UCAV and they are the key information for the training phase of the reinforcement learning agent. The reward function includes detection, tracking, engagement and hit probabilities of the UCAV to generate the required control signal vector for an effective escape maneuver.

## III. RL-Based Path Planning

The reinforcement learning method is developed based on the idea of improving the agent behavior (i.e. policy) by interacting with the environment. The agent tries to maximize its cumulative reward by using the observations from the environment. In a wargame simulation environment, the reward function may be defined by using tracking and hit probabilities of the fleet, fuel consumption, etc. to maximize the mission performance. Similarly, the observation vector for the same environment may contain relative distance angle vectors.

In this study, we used the reinforcement learning method to optimize the flight route plan for the UCAVs in the wargame simulation environment. In the training phase, the RL agent uses the observation vector (i.e. relative geometry, survivability states, aircraft states, etc.) to optimize the policy. In this section, we describe the general structure of the RL agent, the utilized PPO algorithm, observation and action vectors of the UCAV fleet.

### A. Definitions and Assumptions

Prior to explanation of the learning algorithm and RL agent structure, it is important to define the relative geometry between the allied aircrafts and ground based-enemy assets. This information will be used to generate the observation

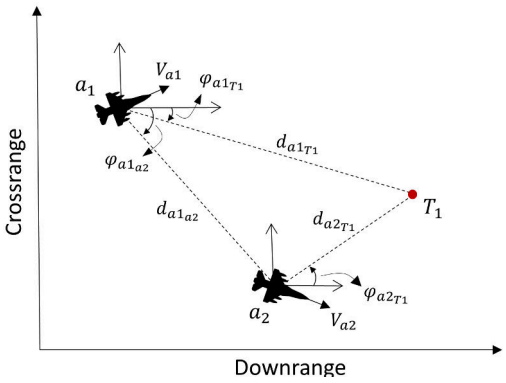


**Fig. 5 Relative geometry between two allied aircrafts and one enemy asset.**

vector which is directly related with the learning performance of the RL agent. The relative geometry between two allied aircrafts and one ground asset is given in Figure (5).

Generalized mathematical definition of the relative geometry is defined in Equations (5 - 8) to solve the problem in the whole wargame environment for $i^{th}$ allied aircraft and $k^{th}$ enemy asset where $i = 1, 2, ..., m$ and $k = 1, 2, ..., n$. Here, $m$ is total number of allied aircrafts and $n$ is total number of enemy assets.

$$d_{a_{iT_k}}(t) = \left\| \mathbf{p}_{a_i}(t) - \mathbf{p}_{T_k} \right\| \tag{5}$$

$$d_{a_{ia_j}}(t) = \left\| \mathbf{p}_{a_i}(t) - \mathbf{p}_{a_j}(t) \right\| \tag{6}$$

$$\varphi_{a_{iT_k}}(t) = atan2\left( \frac{p_{y_{a_i}}(t) - p_{y_{T_k}}}{p_{x_{a_i}}(t) - p_{x_{T_k}}} \right) \tag{7}$$

$$\varphi_{a_{ia_j}}(t) = atan2\left( \frac{p_{y_{a_i}}(t) - p_{y_{a_j}}(t)}{p_{x_{a_i}}(t) - p_{x_{a_j}}(t)} \right) \tag{8}$$

In these equations, $d_{a_{iT_k}}$ is distance between the $i^{th}$ allied aircraft and $k^{th}$ enemy asset, $d_{a_{ia_j}}$ is distance between the $i^{th}$ and $j^{th}$ allied aircraft, $\varphi_{a_{iT_k}}$ is relative angle between the $i^{th}$ allied aircraft and $k^{th}$ enemy asset, $\varphi_{a_{ia_j}}$ is relative angle between the $i^{th}$ and $j^{th}$ allied aircrafts. $\mathbf{p}_{a_i} = [p_{x_{a_i}}, p_{y_{a_i}}]^T$, $\mathbf{p}_{T_k} = [p_{x_{T_k}}, p_{y_{T_k}}]^T$ are position vectors of $i^{th}$ aircraft and $k^{th}$ enemy asset, $V_{a_i}$ is total velocity of the $i^{th}$ allied aircraft.

In an air-to-ground mission, the aircrafts are able to neutralize the enemy assets by using guided and unguided munitions. The unguided munitions, such as MK-82, do not have any guidance system and ballistic trajectory prediction is necessary before release. They also be affected by atmospheric disturbances such as gust and wind which decrease the hit probability of the target. On the other hand, guided munitions, such as GBU-12 and GBU-49, have active control surfaces to guide the munition towards aim point. Hence, some of these guided munitions have circular error probable (CEP) less than 5 m when global positioning system (GPS) is available and they are called *precision guided munitions* (PGM).

In this study, it is assumed that the aircrafts have PGMs that can be used against the ground targets. Also, it is assumed that launch acceptability region (LAR) of the guided munition is sufficiently large and covers the target area when the aircraft pass through the target area boundary. In other words, if the aircraft is in the target area, the guided munition can be released with a high kill probability of the ground target.

### B. Proximal Policy Optimization (PPO) Algorithm

In this study, proximal policy optimization (PPO) algorithm is used in the training phase which attains data efficiency and reliability of thrust region policy optimization (TRPO) while performing first-order optimization process. In addition, an objective function with clipped probability ratios is proposed in this algorithm to form a pessimistic estimate

**Algorithm 1:** PPO Algorithm, Actor-Critic Style [9]

```
for iteration = 1,2,... do
    for actor = 1,2,...N do
        Run policy π_θold in environment for T timesteps;
        Compute advantage estimates Â_1, ..., Â_T;
    end
    Optimize L wrt θ, with K epochs and M ≤ NT;
end
```

of the policy performance [9]. General overview of the PPO algorithm is given in Algorithm 1.

Here, $N$ is number of parallel actors, $T$ is timestep, $K$ is epoch number, $M$ is minibatch size, $\pi_\theta$ is stochastic policy, $\hat{A}_t$ is estimation of the advantage function at timestep $t$, $L$ is surrogate objective, $\theta$ is vector of policy parameters.

### C. Path Planning

For an effective training process and high performance path planning, it is quite critical to define adequate reward function, observation vector and action vector. From the survivability point of view, observation vector should include data from different types of sources to increase the awareness capability of the agent. A feasible and large observation vector provides an efficient learning process and improves the agent performance by defining the input-output relationship clearly. In a typical wargame environment, major data sources may be listed as ground-based radars and airborne radars which provide relative position and velocity information between enemy and allied forces. Also, it is important to have a datalink between the allied forces for an effective coordination during the operation. In the light of these requirements, we designed an observation vector which includes distances and relative angles from allied aircrafts to enemy radars, weapons and target point. In addition, it includes survivability data of the allied aircrafts such as tracking, engagement and hit probabilities. For a seamless coordination between the allied forces, relative position of the allied aircrafts are also measured and shared between each others. General overview of the RL agent and its interactions with the warfare environment is given in Figure (6).

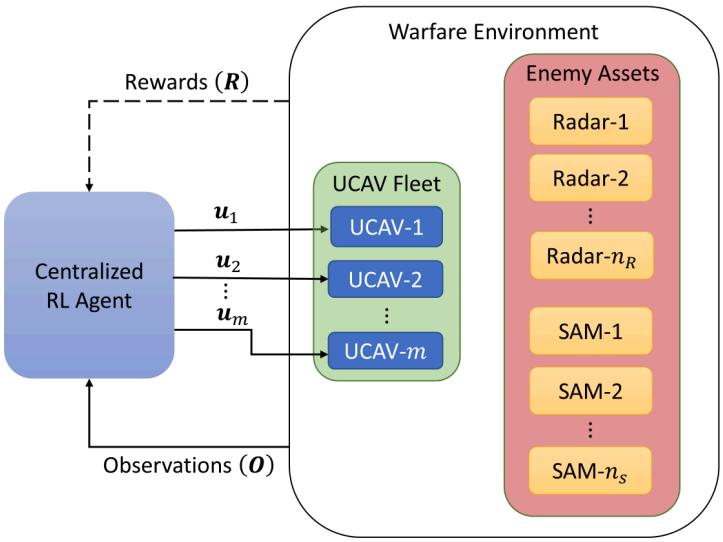


**Fig. 6 General overview of the RL agent and its interaction with the warfare environment.**

As mentioned before, the reward function is developed according to the mission requirements such as minimal detection, tracking, engagement and hit probability levels and collision avoidance. General structure of the reward function for $i^{th}$ aircraft is given in Equation (9).

**Table 1 Reward function parameters.**

| Parameter | Value | Parameter | Value |
|---|---|---|---|
| $k_d$ | -0.1 | $k_{sa}$ | 0.5 |
| $k_t$ | -0.2 | $k_{out}$ | -300 |
| $k_e$ | -0.1 | $k_{stp}$ | -30 |
| $k_h$ | -0.2 | $k_{tg}$ | 50 |
| $k_{ca}$ | -100 | | |

$$R_i = w_{dt_i} R_{dt_i} + w_{eh_i} R_{eh_i} + w_{ca_i} R_{ca_i} + w_{out_i} R_{out_i} + w_{stp_i} R_{stp_i} + w_{t_i} R_{t_i} \tag{9}$$

where $R_{dt_i}, R_{eh_i}, R_{ca_i}, R_{out_i}, R_{stp_i}, R_{t_i} \in \mathbb{R}$ are rewards for detection-tracking, engagement-hit, collision avoidance, being out of the warfare environment, simulation step limitation and reaching to the target area, respectively. Individual weights $w_{dt_i}, w_{eh_i}, w_{ca_i}, w_{out_i}, w_{stp_i}, w_{t_i} \in \mathbb{R}$ span to the multi-objective optimization Pareto-optimal frontier. The reward function elements are defined in Equations (10 - 14).

$$\begin{aligned} R_{dt_i} &= k_d p_{d_i} + k_t p_{t_i} \\ R_{eh_i} &= k_e p_{e_i} + k_h p_{h_i} \end{aligned} \tag{10}$$

$$R_{ca_i} = \begin{cases} k_{ca}, & \text{if distance between } i^{th} \text{ and } j^{th} \text{ UCAVs} < d_{ca} \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

$$R_{out_i} = \begin{cases} k_{out}, & \text{if } i^{th} \text{ UCAV is out} \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

$$R_{stp_i} = \begin{cases} k_{stp}, & \text{if } n_{stp_i} > n_{stp_{max}} \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

$$R_{t_i} = \begin{cases} k_{tg}, & \text{if } i^{th} \text{ UCAV is in the target area} \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

where $p_{d_i}, p_{t_i}, p_{e_i}, p_{h_i} \in \mathbb{R}$ are detection, tracking, engagement and hit probabilities of the $i^{th}$ UCAV, $n_{stp_i}$ is number of the simulation steps in each episode. Minimum allowable distance between the UCAVs is given as $d_{ca} = 100\,m$. Maximum allowable step number of simulation is selected as $n_{stp_{max}} = 1000$. Reward function parameters $k_d, k_t, k_e, k_h, k_{ca}, k_{out}, k_{stp}, k_{tg} \in \mathbb{R}$ are defined as reward values for detection, tracking, engagement, hit, collision avoidance, being out of the map, simulation step limitation and being in the target area, respectively. The value of these parameters are given in Table (1). These values are selected as a result of progressive training, simulation and evaluation studies.

Defining the observation vector is another crucial process before performing the training phase of the agent. A large and feasible observation vector is beneficial to establish a clear relationship between input and output of the system. In the proposed wargame simulation environment, the observation vector $\mathbf{O}$ contains point-mass model states, relative position and orientation data according to the enemy assets and survivability model states of the aircraft fleet. The observation vector of the $i^{th}$ aircraft is given in Equation (15).

$$\mathbf{O}_{a_i} = [\mathbf{p}_{a_i}, \psi_{a_i}, V_{a_i}, \mathbf{d}_{a_{i_{T_k}}}, \mathbf{d}_{a_{i a_j}}, \varphi_{a_{i T_k}}, \varphi_{a_{i a_j}}, \mathbf{p}^T_{s_{a_i}}] \tag{15}$$

where $\mathbf{p}_{s_{a_i}} \in \mathbb{R}^5$ is survivability model states. The total observation vector $\mathbf{O}$ contains the observation data from the whole aircraft fleet and it is given in Equation (16).

$$\mathbf{O} = [\mathbf{O}_{a_1}, \mathbf{O}_{a_2}, ..., \mathbf{O}_{a_m}]^T \tag{16}$$


**Fig. 7 Episode reward of the training process.**

A typical air-to-ground mission with time-of-arrival and no-fly-zone constraints requires two main input for the aircraft platform. These inputs are velocity command ($V_{a_{c_i}}$) and lateral acceleration command ($n_{lat_{c_i}}$) for the $i^{th}$ aircraft. To simplify the problem, closed-loop mathematical model of the lateral acceleration and velocity-hold control systems are defined as first-order dynamics as given in Equation (1). Hence, the proposed centralized RL agent is developed to provide lateral acceleration and velocity commands. The action vector $\mathbf{u} \in \mathbb{R}^{2m}$ is given in Equation (17) where $m$ is number of UCAVs in the allied fleet. To provide a feasible control signal into the system, lateral acceleration and velocity commands are bounded as $n_{lat_{c_i}} \in [-8, +8]\ g$ and $V_{a_{c_i}} \in [250, 350]\ m/s$.

$$\begin{aligned} \mathbf{u} &= [\mathbf{u_1}, \mathbf{u_2}, ..., \mathbf{u_m}]^T \\ &= [n_{lat_{c_1}}, V_{a_{c_1}}, n_{lat_{c_2}}, V_{a_{c_2}}, ..., n_{lat_{c_i}}, V_{a_{c_i}}]^T \end{aligned} \tag{17}$$

## IV. Simulation Results

The training phase of the RL agent is performed throughout 120M steps. To verify the training of the agent, two independent training processes are completed by utilizing the same reward functions and the reward time histories of these training phases are given in Figure (7). As shown in this figure, reward function time histories represent similar characteristics and they converge to reward value which is close to each other. By using the results of the second training process and comparing it with the first one, the repeatibility of the algorithm is verified since the reward values are similar. The training parameters of the PPO algorithm are given in Table (2) for the reproducibility of the application.

The effectiveness of the proposed RL-agent is evaluated by utilizing mission success metrics, which are based on survivability model of the UCAVs, are defined as given in Equation (18) and (19). These success metrics process Monte Carlo simulation results and provide insight about the levels of tracking and hit probabilities of the UCAV fleet.

$$P_T = \frac{1}{q} \sum_{q_c=1}^{q} \left( \frac{1}{m} \sum_{i=1}^{m} \left\| p_{t_{i_{qc}}}(t) \right\|_1 \right) \in [0, 1] \tag{18}$$

$$P_H = \frac{1}{q} \sum_{q_c=1}^{q} \left( \frac{1}{m} \sum_{i=1}^{m} \left\| p_{h_{i_{qc}}}(t) \right\|_1 \right) \in [0, 1] \tag{19}$$

where $P_T, P_H$ are performance metrics for tracking and hit probabilities, $q$ is epoch number of the Monte-Carlo simulation, $m$ is number of UCAVs in the fleet and $p_{t_{i_{qc}}}(t), p_{h_{i_{qc}}}(t)$ are time histories of tracking and hit probabilities of the $i^{th}$ UCAV in the $q_c^{th}$ epoch, $\|.\|_1$ is used to define the 1-norm of the vector. The innermost sum operation defines the total tracking and hit probabilities of the fleet in one simulation episode. Then, mean of the innermost sum operation of the tracking and hit probabilities are calculated by dividing with the total number of UCAVs, $m$. This summation is applied for each episodes in the Monte Carlo simulation and the mean value of the tracking and hit probabilities are calculated by dividing by the number of Monte Carlo simulation episodes, $q$. By utilizing these metrics, one can obtain a normalized mission success level in the interval of $P_T, P_H \in [0, 1]$. For example, if the tracking metric $P_T$ is

**Table 2 Training parameters of the PPO algorithm.**

| Parameters | Value |
|---|---|
| Number of Steps for per update | 128 |
| Number of Minibatches | 4 |
| Discount factor | 0.99 |
| Factor of Bias vs Variance Tradeoff | 0.95 |
| Learning Rate | 5E-5 |
| Cliprange | 0.2 |
| Number of Epoch when minimizing the surrogate | 4 |
| Entropy Coefficient | 0 |

calculated as 0 after the Monte Carlo simulation, it means that any of the agents in the UCAV fleet is not tracked by a surface-based radar in any of the Monte Carlo simulation episodes. If the $P_T$ is calculated as 1, it means that all of the UCAV agents are tracked by the surface-based radars in the each simulation step of the Monte Carlo analysis. A similar example can be given for the mission success metric of the hit probability, $P_H$.

In the first step of the simulation studies, the preliminary results of the training process are evaluated on a single-run simulation with 2 UCAVs in the warfare environment. The generated paths for each UCAV are given in Figure (8) on East-North plane. Here, it is shown that each UCAVs are tend to avoid enemy's SAM-protected areas. They also try to find the path with minimum tracking probability while trying to avoid collision. For this single-run simulation, time history of the survivability states of the UCAVs are given in Figure (9).

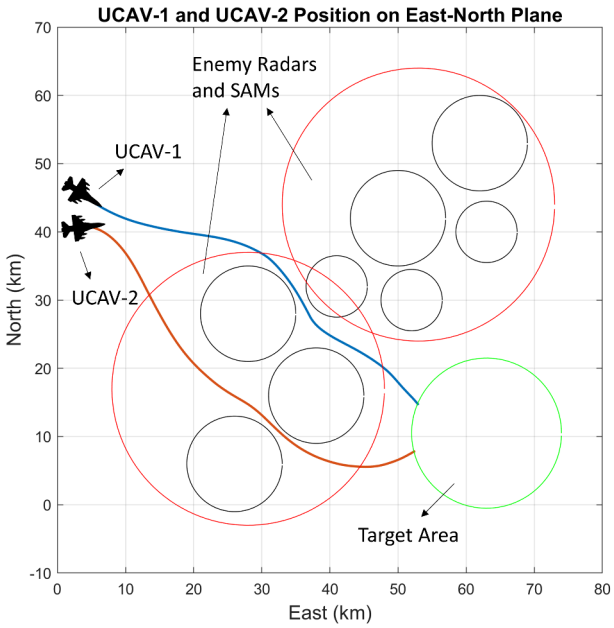


**Fig. 8 Position of the UCAVs on East-North plane.**

After evaluating the results of the single-run simulation, the performance of the RL agent is evaluated in a Monte-Carlo simulation in which starting position and heading attitude of the UCAVs are given randomly. Here, the initial position and heading attitude of the UCAVs are selected in the intervals of $p_e(1) \in [1000, 4000]\ m$, $p_n(1) \in [30000, 60000]\ m$ and $\psi(1) \in [-45°, 45°]$ which are wider than the used intervals in the training phase. The purpose of selecting the wider initial condition intervals is to evaluate the system performance and behavior when the UCAVs are started from an unseen initial position. The comparison of these intervals for training and Monte Carlo analysis is given in Table (3).

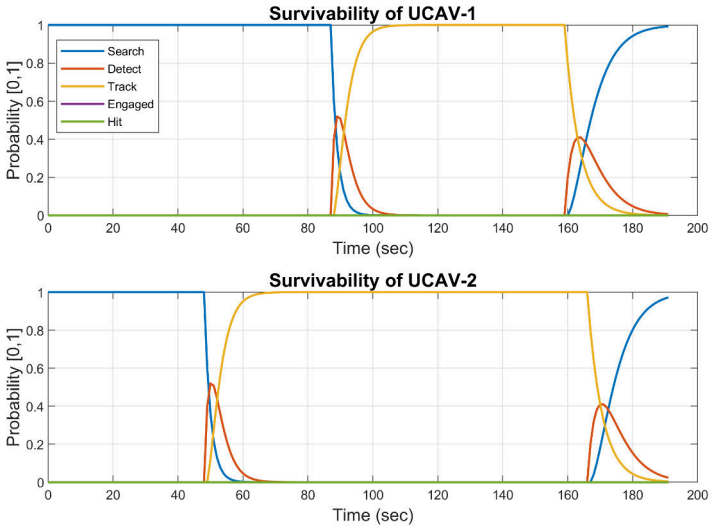


**Fig. 9 Survivabiltiy states of the UCAVs.**

**Table 3 Initial condition intervals of the UCAVs for training phase and Monte Carlo analysis.**

| | $p_n(1)$ (m) | $p_e(1)$ (m) | $\psi(1)$ (deg) |
|---|---|---|---|
| Training | [42000, 47000] | [1000, 4000] | [-45, 45] |
| Monte Carlo | [30000, 60000] | [1000, 4000] | [-45, 45] |

For the clarity of the generated plots, 200-runs Monte-Carlo simulation for UCAV-1 (blue) and UCAV-2 (red) is performed and results are given in Figure (10). The difference between the initialization area of the training phase (green area) and Monte Carlo analysis (area with black boarder line) is also emphasised in this figure. After the 200-runs Monte Carlo simulation which is just for visualization of the flight routes, 2000-runs Monte Carlo analysis is performed to investigate the performance of the agent. Histograms of the $P_T$ and $P_H$ success metric for 2000-runs Monte Carlo simulation are given in Figure (11). Here, $P_H$ histograms are zoomed-in to show the frequency and value of the hit score which are quite low. For a quantitative comparison of the mission success metrics of the UCAV fleet, statistical properties, i.e. mean, mode and median, of the Monte Carlo simulation results are given in Table (4). According to this table, UCAV-1 has lower mean, mode and median for $P_T$ which indicates better radar-avoidance performance for the UCAV-1. On the other hand, UCAV-2 has lower mean for $P_H$ and it has better SAM-avoidance performance when compared to the UCAV-1. Here, it is important to note that the RL agent is able to provide appropriate flight route for the UCAVs that are initialized in a different position which is not experienced in the training phase.

It is interesting to observe that the UCAVs have different mission success metric values although they are guided by the centralized mission planner. This situation is quite clear on $P_T$ since the difference is at about 39.98% between the UCAV-1 and UCAV-2. This can be noted as a future work to evaluate and explain the main reason of the performance difference of the UCAVs which are guided by the same RL agent.

**Table 4 Statistics of the Monte Carlo simulation.**

| | | Mean | Mode | Median |
|---|---|---|---|---|
| UCAV-1 | $P_T$ | 0.4115 | 0.4041 | 0.3836 |
| | $P_H$ | 0.0074 | 0 | 0 |
| UCAV-2 | $P_T$ | 0.5760 | 0.5399 | 0.5896 |
| | $P_H$ | 0.0042 | 0 | 0 |

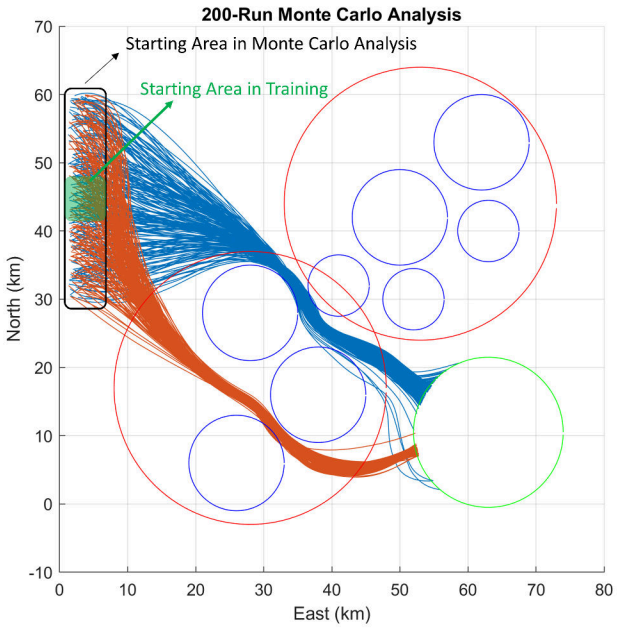


**Fig. 10 200-run Monte Carlo simulation for random initial position and orientation.**

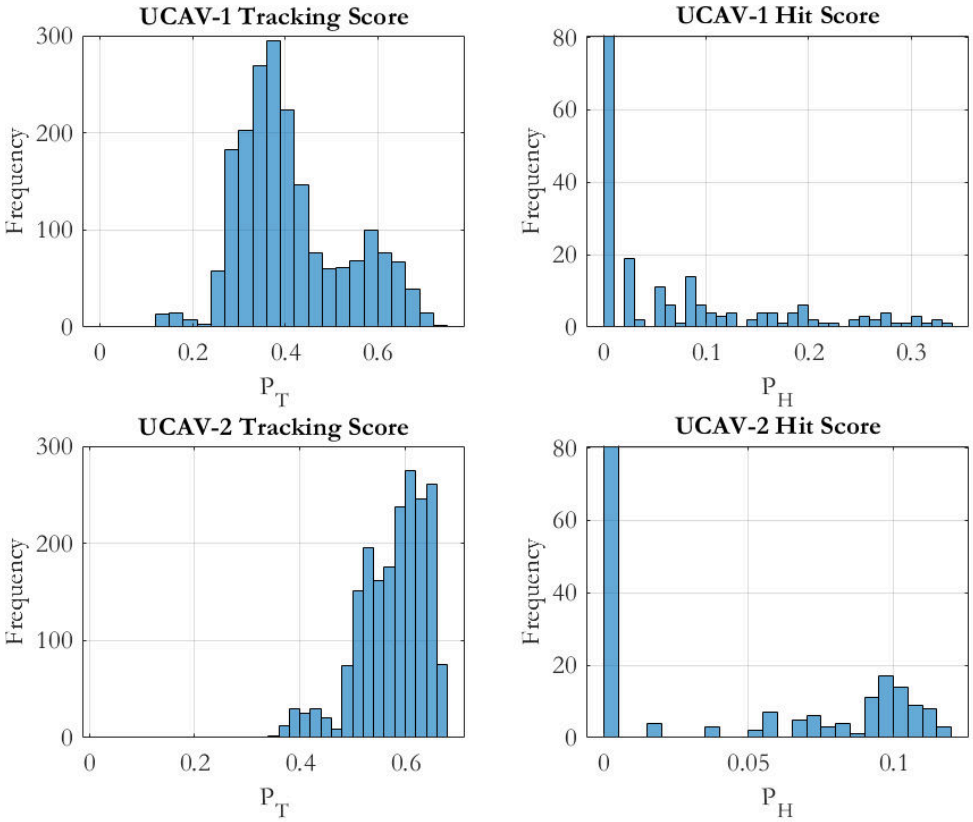


**Fig. 11 $P_T$ and $P_H$ histograms of 2000-run Monte Carlo simulation with random initial position and orientation.**

## V. Conclusion

This paper deals with the autonomy development for a UCAV fleet in the warfare simulation environment by utilizing the reinforcement learning method. Flight path planning of the fleet is performed by combining the survivability model, mission success metrics and PPO approach. Air-to-ground operation is defined as flying through the enemy air defense systems, infiltrating into the target area and neutralizing it. Mission success is determined as a function of collision avoidance and survival probability of the aircraft fleet. Aircrafts are modeled in point mass concept with acceleration and velocity command inputs. 5-State survivability model is utilized which includes searching, detected, tracking, engaged and hit states. It provides survival probability of the aircraft during the mission. An RL agent is generated which has actor-critic structure. Observation state vector and reward function are developed and used in the training phase of the RL agent. Then, the generated agent is trained by utilizing the PPO algorithm to maximize the total reward of the UCAV fleet.

Simulation studies are performed in the form of single-run and Monte Carlo analysis. The single-run simulation results indicate that the trained RL agent is able to guide the UCAV fleet while avoiding the enemy SAMs. The time histories of the 5-state survivability model is evaluated in this step. For a further analysis of the proposed path planning algorithm, 2000-runs Monte Carlo simulation is performed with random initial conditions for a given position and heading attitude interval. Wider initial condition intervals are utilized to evaluate the system performance for the initial conditions which are not experienced in the training phase. The system performance is examined based on the designed mission success metrics, i.e. $P_T$ and $P_H$. It is shown that the trained RL agent is able to generate appropriate flight routes for the UCAVs even if they are not started in the initial conditions experienced in the training phase. This gives preliminary insight about the generalization ability of the RL agent. In this analysis, it is also observed that $P_T$ and $P_H$ metrics for the UCAVs are different even though they are guided by the centralized RL agent and its explanation can be noted as a possible future work.

In future works, we will be further detailing the RL process and providing scenarios showing the capability of the approach for generating structured coordinated attack strategies. Also, target kill probability model and multiple-shoot capability of the enemy weapon systems will be developed and integrated into the warfare environment to obtain the whole wargame simulation. The LAR of the air-to-ground weapon will be integrated into the simulation environment and the required weapon launch conditions will be detailed. In addition, fuel consumption will be modeled to introduce the cost-effectiveness while generating the flight routes with high survival probabilities. Also, the generalization ability of the RL agent will be further investigated by evaluating the system performance for a wide range of initial conditions which are not seen in the training phase.

## Acknowledgement

The authors would like to thank Mustafa Demir at Istanbul Technical University Aerospace Research Center for his support and advice on implementation issues in Python.

## References

[1] Ball, R. E., *The fundamentals of aircraft combat survivability: analysis and design*, American Institute of Aeronautics and Astronautics, 2003.

[2] Erlandsson, T., and Niklasson, L., "A five states survivability model for missions with ground-to-air threats," *Modeling and Simulation for Defense Systems and Applications VIII*, Vol. 8752, International Society for Optics and Photonics, 2013, p. 875207.

[3] Erlandsson, T., and Niklasson, L., "Comparing air mission routes from a combat survival perspective," *The Twenty-Sixth International FLAIRS Conference*, 2013.

[4] Erlandsson, T., and Niklasson, L., "Automatic evaluation of air mission routes with respect to combat survival," *Information Fusion*, Vol. 20, 2014, pp. 88–98.

[5] Wang, X., Song, B.-F., and Hu, Y.-F., "Analytic model for aircraft survivability assessment of a one-on-one engagement," *Journal of aircraft*, Vol. 46, No. 1, 2009, pp. 223–229.

[6] Hui, L., and Wang, K. M., "Torpedo performance Markov model," *Expert Systems with Applications*, Vol. 42, No. 23, 2015, pp. 9129–9136.

[7] Kim, M.-H., Lee, H.-M., Wei, Y., and Lee, M.-C., "A study of new path planning algorithm using extended a* algorithm with survivability," *International Conference on Intelligent Robotics and Applications*, Springer, 2012, pp. 608–617.

[8] Baspinar, B., and Koyuncu, E., "Survivability Based Optimal Air Combat Mission Planning with Reinforcement Learning," *2018 IEEE Conference on Control Technology and Applications (CCTA)*, IEEE, 2018, pp. 664–669.

[9] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O., "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

2021-01-04

# Development of UCAV fleet autonomy by reinforcement learning in a wargame simulation environment

Yuksek, Burak

AIAA

Yuksek B, Umut Demirezen M, Inalhan G. (2021) Development of UCAV fleet autonomy by reinforcement learning in a wargame simulation environment. In: AIAA SciTech Forum 2021, 11-15 and 19-21 January 2021, Online

https://doi.org/10.2514/6.2021-0175